cups-filters

The non-Apple part of CUPS, maintained by OpenPrinting

ippusbxd

Standards-conforming support for IPP-over-USB printers

Till Kamppeter, OpenPrinting

THE LINUX FOUNDATION

Introduction

- **cups-filters takes up everything from CUPS which Mac OS X does not need (CUPS 1.6.x)**
  - Started end of 2011 by OpenPrinting, overtaking most of CUPS' **filters**
  - Switched filters over from PostScript-centric to **PDF-centric** workflow
  - **cups-browsed** introduced end of 2012, to introduce browsing of DNS-SD-advertised remote CUPS queues, as CUPS dropped its own broadcasting/browsing
- **In 7 years of development cups-filters improved a lot**
  - Auto-create print queues for **IPP network** and **IPP-over-USB** printers, especially **driverless** printing, not competing with **CUPS' own temporary queues**.
  - **Mobile** printing support: No printer setup tool needed, auto-setup of printers, driverless, cups-browsed auto-shutdown
  - Load-balanced **printer clusters**, with client-side or server-side queueing of jobs
  - Do **legacy CUPS broadcasting/browsing** to work with old CUPS on remote machines
  - Filters support **Ghostscript**, **Poppler**, and **MuPDF** as PDF interpreter
  - Filters support all PDLs for **driverless** printing: **PDF**, **Apple Raster**, **PWG Raster**, and **PCLm**

Role of cups-browsed

- **CUPS auto-generates its own temporary queues to point to remote IPP/CUPS printers, why continue cups-browsed?**
  - **Printer clustering** (like the former Implicit Classes)
    - Configurable: Automatic (by equal remote queue names) and manual (by cups-browsed.conf)
    - Load balancing of clustered queues
    - Auto-selection of destination printer by job and job settings (planned)
  - **Fine-grained filtering** of which remote printers are available
    - Not only servers/IP addresses, but also service names. PDLs, color, duplex, ...
    - When using CPDB CUPS backend turn off display of temporary CUPS queues
  - Support for **new driverless printing technologies** not (yet) supported by CUPS
    - Currently: PCLm
  - **Legacy technology** support
    - Legacy CUPS server (1.5.x and older) interoperability
    - **Interoperability with old print dialogs** (which do not use cupsEnumDests() and new CUPS API, nor CPDB)
    - Support for legacy (IPP 1.x) IPP network printers

New features in cups-filters: cups-browsed

- **Support for CUPS' own temporary queues for remote IPP/CUPS printers**
  - User can configure to not create a local queue for a remote printer for which CUPS would create a temporary queue, or create a local queue, with the PPD generated by CUPS or by cups-filters
  - Make queue permanent when overwriting a temporary CUPS queue
- **More configurable printer clustering**
  - Configurable **queue naming** scheme: Remote queue name, make/model, DNS-SD service name
  - **Automatic clustering** can be turned off
  - **Manual clustering** via cups-browsed.conf possible
- **For Printer Applications (including ippusbxd for IPP-over-USB)**
  - Configuration option to only create local queues for IPP printers on localhost
- **Support for all driverless printing standards**
  - Support for **PCLm** (for **Mopria** and **Wi-Fi Direct**) as the last of the 4 PDLs (PDF, Apple Raster, PWG Raster, PCLm

New features in cups-filters: cups-browsed

- **PPD file generator**
  - Keeping up with the functionality of **CUPS' PPD generator**
  - Make sure that all options and choices (including paper sizes) have **human-readable strings** (from CUPS' translation tables and from the printer).
  - **Resolution**: Make sure there are no unreasonable ones (from printer firmware bugs), make sure resolution selection for low, normal, and high print quality is reasonable.
  - **PCLm** support
- **Robustness of generated queues against user error**
  - **Deleted** queues get automatically re-created
  - **Overwritten** queues get released from cups-browsed control and, if possible, re-created with new name
- **Bug fixes/reliability improvements**
  - **Remote CUPS queues are recognized** by DNS-SD TXT record, not by URI (HP LaserJet Professional M1212nf MFP has CUPS-typical URI)
  - **Crash bug fixes**: Removed broken comparison function for remote printer CUPS array, more NULL checks, initialization of data structures, ...
  - More verbose **debug logging**

New features in cups-filters: Filters

- **pdftoopvp and pdftoijs deprecated**
  - Only built if requested via ./configure command line. With this it should be tested whether they are actually still used by someone. If no one complains they will later get completely removed.
- **Support for the PCLm output format**
  - Driverless printing on **Mopria** and **Wi-Fi Direct** printers
  - Google Summer of Code 2017 project of Sahil Arora.
- **Flattening of interactive PDF forms (workaround)**
  - Flatten PDF with interactive form to static PDF so that further manipulation, like scaling, number-up, ... do not let the filled form content getting lost.
  - Implemented by using pdftocairo of Poppler and if this fails Ghostscript
  - Will probably be replaced by a QPDF-based solution later.
  - Options on Ghostscript calls for more reliable PDF form printing.
- **Bug fixes:**
  - Handling errors and missing utilities better
  - prettyprint

New features in cups-filters: General

- **Moved repository to GitHub**
  - Related projects, like CUPS, use GitHub
  - Many Contributors and GSoC students use GitHub
  - Pull requests, Issues, Releases, …
  - Possibility to create project web sites easily
- **Clean-up and fixes in the build system**
  - "make dist" works correctly now, not depending on system
  - .gitignore
- **Eliminated all compiler warnings**
- **Compatibility with Poppler 0.58.0**
- **Documentation updates and fixes**

Future of cups-filters

- **Make cups-browsed re-startable in-process**
  - For example to re-read its configuration and restart on "kill -HUP" (like most other daemons).
- **Let cups-browsed not use CUPS PPD APIs any more**
  - Do not download PPDs from remote CUPS printers
- **Let cups-browsed treat IPP network printers and remote CUPS printers equal**
  - Allow clustering of any combination of printers.
- **Make cups-browsed auto-select printers in a cluster of very different printers depending on the job and the option settings supplied by the user**
- **QPDF-based solutions for bannertopdf and form-flattening**
  - Remove dependency on Poppler
- **Provide infrastructure (library functions, ...) for Printer Applications?**
- **Translate PPDs into languages supported by CUPS' translations tables.**

ippusbxd – IPP-over-USB daemon

- **ippusbxd development**
  - Major changes in TCP and USB communication David Valleau (Google)
    - Keep all USB interfaces claimed to avoid bad effects when releasing one while another is communicating.
    - Eliminate the need of parsing the HTTP stream, by having one thread for each data direction
      - Reduces amount of code
      - Allows support for IPPS
  - Support for the PDL PCLm in the DNS-SD record
  - Use URF field in the USB device ID for the DNS-SD record
  - Use default port 60000 and connect to the first discovered printer by default
    - One printer connected, and simple "ippusbxd" call -> ipp://localhost:60000/ipp/print
  - Updated Avahi patch in readme.md
- **Features for the coming year**
  - IPPS support (David Valleau?)
  - Fix overall slowness found in the web interface, due to many parallel requests from the browser for which there are not enough USB interfaces (David Valleau)
  - Re-use code for Printer Applications (library?)

Avahi

- **Needs to support services on localhost (loopback device "lo")**
  - For **ippusbxd** and **Printer Applications** in general
  - **Patch** is available
    - Simple changes, done by me and Rithvik Patibandla (GSoC 2017 student)
    - Submitted upstream as pull request
    - Maintainer Trent Lloyd did not answer
- **Avahi can be considered unmaintained. Volunteers?**

Printer Applications

- **Framework for Printer Applications (multi-student GSoC 2019 project?)**
  - Printer Application: Daemon like ippusbxd, emulating a driverless IPP printer, running input data through driver for printer's PDL and to printer via IPP, Socket, USB, ...
  - Wrap legacy drivers (Foomatic, HPLIP, Gutenprint, foo2zjs, ...) into Printer Applications: Universal CUPS driver wrapper Printer Application?
  - Change driver design guidelines for manufacturers (to create Printer Application Snaps)
  - Snap all Printer Applications and put in Snap Store
  - Printing Stack Snap DOES NOT need a driver interface any more
  - LSB Printer driver packages are deprecated
- **Will CUPS backends (except IPP) be deprecated?**
  - Should be turned into library functions for Printer Applications.
- **ippusbxd is a special form of a Printer Application**
  - Pass-through, as no PDL conversion needed, only connection type conversion.
- **Printer Applications: ipp(s)://localhost:<port>/...,**
  - Same Avahi changes as for ippusbxd needed

Questions?

cups-filters – Original ideas

- **From version 1.6.x on CUPS dropped features not needed for Mac OS X**
  - Filters for file conversion and Postscript workflow: imagetops, pdftops, pstoraster, …
  - Serial and parallel backends
  - CUPS broadcasting/browsing for automatic availability of shared printers on remote CUPS client, replacement technology DNS-SD has only broadcasting and no browsing, also incompatible with old CUPS versions
- **Dropped CUPS filters were put into there own source package on CUPS SVN → Seed for cups-filters**
- **cups-filters started by OpenPrinting end of 2011**
  - Legacy filters package of CUPS with pure PS workflow filters dropped
  - PDF workflow filters (formally maintained as CUPS add-on) added
  - cups-browsed added end of 2012 for DNS-SD browsing and legacy CUPS broadcasting/browsing

The Filters and Backends

- **Dropped CUPS filters made up a PostScript-based print workflow**
- **cups-filters supplies filters for the new PDF-based print workflow**
  - Filters from CUPS legacy package: commandtoescpx, commandtopclx, imagetoraster, pdftops, rastertoescpx, rastertopclx
  - PDF filters from OpenPrinting Japan: pdftopdf (old Poppler-based), pdftoraster (Poppler-based), pdftoopvp (deprecated), imagetopdf
  - Filters from Google Summer of Code projects: texttopdf, pdftoijs (deprecated), pdftopdf (new QPDF-based), rastertops, mupdftoraster, rastertopclm
  - Also added: bannertopdf, texttotext, gstopdf, rastertopdf, gstoraster, gstopxl, foomatic-rip, filters for Braille embossers
  - Dropped filters from CUPS legacy package: bannertops, imagetops, texttops
- **cups-filters adopts serial and parallel backends**
- **cups-filters used with Ghostscript, Poppler, or MuPDF as PDF interpreter, Ghostscript allows also PS input**

cups-browsed

- **CUPS 1.6.x drops CUPS broadcasting and browsing**
- **CUPS 1.6.x does DNS-SD broadcasting as defined as PWG standard, but does no DNS-SD browsing**
- **cups-browsed solves the problems caused by this (works with all apps):**
  - In **default configuration**, cups-browsed does **DNS-SD browsing** and auto-generates local queues pointing to the discovered remote CUPS queues, solving the problem with remote CUPS 1.6.x servers
  - **Legacy Option** (via config file): cups-browsed does **CUPS browsing** on CUPS 1.6.x clients with older remote CUPS servers
  - **Legacy Option:** cups-browsed does **CUPS broadcasting** on CUPS 1.6.x servers for older remote CUPS clients.
  - **Legacy Option:** cups-browsed does **BrowsePoll**, on CUPS 1.6.x clients for servers with broadcasting turned off.
- **Alternative: Adding DNS-SD browsing to the print dialog (GUI apps using this dialog only!)**

Let's go mobile!

- **Mobile systems have different printing system demands:**
  - Move between different local networks (home, office, …)
  - No local printers, only network printers
  - Simple UI, no printer setup tool
  - Lightweight printing stack → No driver/PPD library
  - Save battery power, avoid permanently running daemons
- **cups-browsed browses DNS-SD broadcasts:**
  - It picks up remote CUPS queues
  - It picks up network-connected printers:
    - Only IPP printers, as they provide capability info
    - IPP Everywhere, AirPrint, Mopria, Wi-Fi Direct driverless (PDF or raster formats, excellent capability info)
    - Other known PDLs: PostScript, PCL 5c/5e/6/XL
    - Auto-generate PPD, create queue
  - Removes queues on shutdown or when printer disappears
- **cups-browsed optionally shuts down automatically when not needed any more**